



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/047,011	01/16/2002	Lorin Ullmann	AUS920010750US1	6350

7590 06/06/2005  
Robert H. Frantz  
P.O. Box 23324  
Oklahoma City, OK 73123-2334

EXAMINER
----------

MITCHELL, JASON D

ART UNIT	PAPER NUMBER
----------	--------------

2193

DATE MAILED: 06/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

10/047,011

Applicant(s)

ULLMANN ET AL.

Examiner

Jason Mitchell

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 07 April 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-18 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-18 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

1. This action is in response to remarks filed on 4/6/05.
2. As per Applicant's request Claims 1-2, 7, and 13-15 were amended, and claims 16-18 were added. Claims 1-18 are pending in this case.

### ***Response to Arguments***

3. Applicant's arguments with respect to claims 1-15 have been considered but are moot in view of the new ground(s) of rejection.

### ***Claim Objections***

Applicant's amendments were sufficient to overcome the objection, which is consequently withdrawn.

### ***Claim Rejections - 35 USC § 101***

Applicant's amendments were sufficient to overcome the rejection, which is consequently withdrawn.

### ***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

**5. Claims 1-3 and 7-9 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,950,003 to Kaneshiro et al. (Kaneshiro) in view of US 6,807,583 to Hrischuk et al. (Hrischuk).**

**Regarding Claims 1 and 7:** Kaneshiro discloses a method and computer program product for marking a processing stack with signatures to indicate which portions of the stack were utilized by one or more software code modules (col. 7, lines 55-57 'collects profile data at the procedure level'), said method comprising the steps of: inserting stack signing software into one or more code modules stored in a computer-readable medium (col. 11, lines 25-28 'an instruction insertion phase'); producing one or more executable programs containing one or more executable code modules containing said inserted stack signing software (col. 3, lines 53-55 'during compilation'); upon execution of said executable code modules, assigning unique module identifier values to said code modules by said stack signing software (col. 18, lines 46-50 'caller procedure ID'), and pushing onto a processing stack said stack signatures within stack frames allocated to said code modules (col. 18, lines 46-50 'caller-callee information is pushed into the stack area').

Kaneshiro does not explicitly disclose applying his invention to an object-oriented programming language, but does disclose that his invention 'is applicable to cases where other languages are used' (col. 24, lines 5-9).

Hrischuk teaches a method of instrumenting multiple instances of an object in an object-oriented language (col. 23, lines 19-22 'If an object-oriented system is being monitored then the object identifier should include ... instance number') in an analogous art for the

purpose of monitoring the execution of a program (col. 23, lines 19-22 'If an object-oriented system is being monitored').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to apply Kaneshiro's invention to an object oriented language as taught in Hrischuk (col. 23, lines 19-22) in such a way as to prevent module identifiers from having a same value for multiple instances of any re-entered or multiply instantiated code module (Hrischuk col. 23, lines 19-22 'instance number') because one of ordinary skill in the art would have been motivated to broaden the range of program languages to which Kaneshiro's invention was applicable (col. 24, lines 5-9).

**Regarding Claims 2 and 8:** The rejections of claims 1 and 7 are incorporated, respectively; further, Kaneshiro discloses pushing onto said processing stack (col. 18, lines 46-50 'pushed into the stack area') unique module identifier values (col. 18, lines 46-50 'caller procedure ID'), but does not explicitly address object instances.

Hrischuk teaches generating an instance count for each instantiation of an object (col. 23, lines 19-22 'instance number').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to include Hrischuk's instance count (col. 23, lines 19-22 'instance number') as part of Kaneshiro's stack unique module identifier (col. 18, lines 46-50 'caller procedure ID') to maintain its uniqueness.

**Regarding Claims 3 and 9:** The rejections of claims 1 and 7 are incorporated, respectively; further, Kaneshiro discloses pushing onto said stack an entry/exit indicator

associated with said unique module identifier (col. 18, lines 46-50 'Upon entering a procedure ... information is pushed into the stack area').

**6. Claims 4-6 and 10-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,950,003 to Kaneshiro et al. (Kaneshiro) in view of US 6,807,583 to Hrischuk et al. (Hrischuk) further in view of US 6161219 to Ramkumar et al. (Ramkumar).**

**Regarding Claims 4 and 10:** The rejections of claims 1 and 7 are incorporated, respectively; further Kaneshiro discloses inserting stack signature marking software segments into application source code (col. 13, lines 23-24 "start procedure" and "end procedure" are inserted'), Kaneshiro does not disclose inserting the instrumentation codes prior to compilation, in order to map between original and transformed code (col. 3, lines 55-57 'mapping between an original code and a transformed code').

Ramkumar teaches insertion of instrumentation instructions prior to compilation (col. 10, lines 35-37 'pre-compiler instruments the source program') in an analogous art for the purpose of implementing 'portable checkpointing based on source-to-source pre-compilation' (col. 10, lines 49-50).

It would have been obvious to a person of ordinary skill in the art at the time of the invention to insert the instrumentation instructions disclosed in Kaneshiro (col. 3, lines 53-55 'inserting profiling instrumentation codes') prior to compilation as taught in Ramkumar (col. 10, lines 35-37), if it was known that the code would not be transformed, as disclosed in Kaneshiro (col. 3, lines 55-57 'mapping between an original

Art Unit: 2193

code and a transformed code'), because one of ordinary skill in the art would have been motivated to make the instrumentation portable as taught in Ramkumar (col. 10, lines 49-50).

**Regarding Claims 5, 11 and 14:** The rejections of claims 4, 10 and 13 are incorporated, respectively; further Kaneshiro discloses providing a global control (col. 8, Table 1 'start profile (name, record) ... Called at the beginning of the main subroutine') which indicates all application source code modules are to have stack signature marking software segments inserted into them during a given compilation job (col. 3, lines 53-55 'inserting profiling instrumentation codes during compilation').

**Regarding Claims 6, 12 and 15:** The rejections of claims 4, 10 and 13 are incorporated, respectively; further Kaneshiro discloses providing a selective control (col. 8, lines 51-52 "'start log if' is used for starting profiling') which indicates only certain application source code modules are to have stack signature marking software segments inserted into them during a given compilation job (col. 3, lines 53-55 'inserting profiling instrumentation codes during compilation').

**Regarding Claim 13:** Kaneshiro discloses a system for inserting stack signature marking code segments into application software modules in cooperation with a compiler (col. 3, lines 53-55 'inserting profiling instrumentation codes during compilation') and comprising: a control means operable by a user to indicate whether or not to insert stack signature marking code segments into application software modules (col. 8, lines 52-53 "'start log if' is used for starting profiling'); and a code insertion means which, responsive to the operation of the control means, searches for entry

points and exits points in application software modules and inserts stack signature marking code segments following each entry point and prior to each exit point into said application software modules (col. 13, lines 23-24 “start procedure” and “end procedure” are inserted’); a compiler means for producing one or more executable programs (col. 3, lines 53-55 ‘during compilation’) containing one or more executable code modules containing said inserted stack signing software (col. 11, lines 25-28 ‘an instruction insertion phase’); and a debugger means configured to, upon execution of said executable code modules, assign unique module identifier values to said code modules by said stack signing software (col. 18, lines 46-50 ‘caller procedure ID’).

Hrischuk teaches a method of instrumenting multiple instances of an object in an object-oriented language (col. 23, lines 19-22 ‘If an object-oriented system is being monitored then the object identifier should include ... instance number’) in an analogous art for the purpose of monitoring the execution of a program (col. 23, lines 19-22 ‘If an object-oriented system is being monitored’) in an analogous art for the purpose of monitoring the execution of a program (col. 23, lines 19-22 ‘If an object-oriented system is being monitored’).

Ramkumar teaches insertion of instrumentation instructions prior to compilation (col. 10, lines 35-37 ‘pre-compiler instruments the source program’) in an analogous art for the purpose of implementing ‘portable checkpointing based on source-to-source pre-compilation’ (col. 10, lines 49-50).

It would have been obvious to a person of ordinary skill in the art at the time of the invention to apply Kaneshiro’s invention to an object oriented language as taught in



Art Unit: 2193

Hrischuk (col. 23, lines 19-22) in such a way as to prevent module identifiers from having a same value for multiple instances of any re-entered or multiply instantiated code module (Hrischuk col. 23, lines 19-22 'instance number') because one of ordinary skill in the art would have been motivated to broaden the range of program languages to which Kaneshiro's invention was applicable (col. 24, lines 5-9).

Further, it would also have been obvious to a person of ordinary skill in the art at the time of the invention to insert the instrumentation instructions disclosed in Kaneshiro (col. 3, lines 53-55 'inserting profiling instrumentation codes') prior to compilation as taught in Ramkumar (col. 10, lines 35-37), if it was known that the code would not be transformed, as disclosed in Kaneshiro (col. 3, lines 55-57 'mapping between an original code and a transformed code'), because one of ordinary skill in the art would have been motivated to make the instrumentation portable as taught in Ramkumar (col. 10, lines 49-50).

**7. Claims 16 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,950,003 to Kaneshiro et al. (Kaneshiro) in view of US 6,807,583 to Hrischuk et al. (Hrischuk) further in view of 6,862,696 to Voas (Voas).**

**Regarding Claim 16 and 17:** The rejections of claims 1 and 2 are incorporated, respectively; further Kaneshiro does not disclose encrypting said stack signature.

Voas discloses encrypting test data (col. 7, lines 28-31 'Instrumented Version 106 may include encryption') in an analogous art for the purpose of safely transferring said data

to a test lab (col. 7, lines 28-31 'so that data transferred ... protect such data from outside monitoring').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to encrypt the data collected by Kaneshiro's invention (col. 7, line 66 'Three types of information are collected') thereby inherently encrypting the stack signature, because one of ordinary skill in the art would have been motivated to 'test the application in a new market' (Voas col. 7, lines 8-15).

**8. Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,950,003 to Kaneshiro et al. (Kaneshiro) in view of US 6,807,583 to Hrischuk et al. (Hrischuk) and further in view of US 6,519,767 to Carter et al. (Carter).**

**Regarding Claim 18:** The rejection of claim 1 is incorporated; further, Kaneshiro does not explicitly disclose applying his invention to an object-oriented programming language, but does disclose a unique module identifier (col. 18, lines 46-50 'caller procedure ID').

Carter teaches generating a pseudo-random identifier (col. 12, lines 49-54 'random generation routine used ... to generate these identifiers') in an analogous art for the purpose of generating unique object identifiers (col. 12, lines 49-54 'no two identifiers ... are identical').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to use Carter's method of generating identifiers (col. 12, lines 49-54) to generate the module identifiers disclosed in Kaneshiro (col. 18, lines 46-50 'caller

Art Unit: 2193

procedure ID') because one of ordinary skill in the art would have been motivated to generate an identifier which would be unique (col. 12, lines 49-54 'no two identifiers ... are identical').

### ***Conclusion***

9. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. US 6,539,339 discloses profiling program execution based on the program stack. US 5,606,661 discloses identifying objects using randomly generated numbers.

Art Unit: 2193

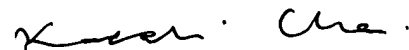
Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571) 272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571) 272-3719. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Jason Mitchell  
5/17/05



**KAKALI CHAKI**  
**SUPERVISORY PATENT EXAMINER**  
**TECHNOLOGY CENTER 2100**